

AMENDMENTS TO THE CLAIMS

The text of all pending claims, (including withdrawn claims) is set forth below. Cancelled and not entered claims are indicated with claim number and status only. The claims as listed below show added text with underlining and deleted text with ~~striketrough~~. The status of each claim is indicated with one of (original), (currently amended), (cancelled), (withdrawn), (new), (previously presented), or (not entered). Please AMEND claims 16-24 in accordance with the following:

Claims 1-15 (Cancelled).

16. (CURRENTLY AMENDED) A method for controlling a plurality of threads that perform parallel processing, comprising:

~~monitering-counting~~ a number of running threads performing parallel processing using a running thread counter and a number of standby threads that are in a standby state using a standby thread counter for a predetermined time period;

setting a maximum number of running threads in accordance with the number of running threads during the predetermined time period using a maximum running thread counter;

comparing the number of standby threads with the maximum number of running threads;
and

terminating a number of standby threads exceeding the maximum number when the number of standby threads is greater than the maximum number;

incrementing the number of running threads using the running thread counter and decrementing the number of standby threads using the standby thread counter in response to a run request; and

updating the maximum number of running threads set by the maximum running thread counter if the incremented number of running threads is greater than the maximum number of running threads.

17. (CURRENTLY AMENDED) A method for controlling a plurality of threads that perform parallel processing, comprising:

~~monitering-counting~~ a number of running threads performing parallel processing using a running thread counter and a number of standby threads that are in a standby state using a standby thread counter for a predetermined time period;

setting an average number of running threads in accordance with the number of running threads during the predetermined time period using an average running thread counter;

comparing the number of standby threads with the average number of running threads;
and

terminating a number of standby threads exceeding the average number when the number of standby threads is greater than the average number;

incrementing the number of running threads using the running thread counter and decrementing the number of standby threads using the standby thread counter in response to a run request; and

updating the average number of running threads set by the average running thread counter if the incremented number of running threads is greater than the average number of running threads.

18. (CURRENTLY AMENDED) A method for controlling a plurality of threads that perform parallel processing, comprising:

~~monitoring-counting~~ a number of running threads performing parallel processing using a running thread counter and a number of standby threads that are in a standby state using a standby thread counter for a predetermined time period;

setting a product obtained by multiplying the number of running threads during the predetermined time period by a predetermined coefficient using a product running thread counter;

comparing the number of standby threads with the product; ~~and~~

terminating a number of standby threads exceeding the product when the number of standby threads is greater than the product;

incrementing the number of running threads using the running thread counter and decrementing the number of standby threads using the standby thread counter in response to a run request; and

updating the product set by the product running thread counter if the incremented number of running threads is greater than the product.

19. (CURRENTLY AMENDED) A controller for controlling a plurality of threads that perform parallel processing, comprising:

a thread management table storing thread information of the plurality of threads, wherein the thread information includes a number of running threads performing parallel processing and

a number of standby threads that are in a standby state, wherein the thread management table includes

a running thread counter to count the number of running threads,

a standby thread counter to count the number of standby threads, and

a maximum running thread counter;

a thread management circuit requesting thread generation based on the number of standby threads stored in the thread management table, and requesting a standby thread to run;

a comparison circuit setting a maximum number of running threads during a predetermined period in accordance with the number of running threads included in the thread information to the maximum running thread counter, and comparing the number of standby threads with the maximum number of running threads; ~~and~~

a termination circuit terminating a number of standby threads exceeding the maximum number when the number of standby threads is greater than the maximum number; and

a thread to cause the running thread counter to increment the number of running threads and the standby thread counter to decrement the number of standby threads in response to a run request, wherein the thread updates the maximum number of running threads set by the maximum running thread counter if the incremented number of running threads is greater than the maximum number of running threads.

20. (CURRENTLY AMENDED) A controller for controlling a plurality of threads that perform parallel processing, comprising:

a thread management table storing thread information of the plurality of threads, wherein the thread information includes a number of running threads performing parallel processing and a number of standby threads that are in a standby state, wherein the thread management table includes

a running thread counter to count the number of running threads,

a standby thread counter to count the number of standby threads, and

an average running thread counter;

a thread management circuit requesting thread generation based on the number of standby threads stored in the thread management table, and requesting a standby thread to run;

a comparison circuit setting an average number of running threads during a predetermined time period in accordance with the number of running threads included in the thread information to the average running thread counter, and comparing the number of standby threads with the average number of running threads; ~~and~~

a termination circuit terminating a number of standby threads exceeding the average number when the number of standby threads is greater than the average number; and
a thread to cause the running thread counter to increment the number of running threads
and the standby thread counter to decrement the number of standby threads in response to a
run request, wherein the thread updates the average number of running threads set by the
average running thread counter if the incremented number of running threads is greater than the
average number of running threads.

21. (CURRENTLY AMENDED) A controller for controlling a plurality of threads that perform parallel processing, comprising:

a thread management table storing thread information of the plurality of threads, wherein the thread information includes a number of running threads performing parallel processing and a number of standby threads that are in a standby state, wherein the thread management table includes

a running thread counter to count the number of running threads,

a standby thread counter to count the number of standby threads, and

a product running thread counter;

a thread management circuit requesting thread generation based on the number of standby threads stored in the thread management table, and requesting a standby thread to run;

a comparison circuit setting a product to the product running thread counter, the product being obtained by multiplying the number of running threads during a predetermined time period by a predetermined coefficient in accordance with the number of running threads included in the thread information, and comparing the number of standby threads with the product; and

a termination circuit terminating a number of standby threads exceeding the product when the number of standby threads is greater than the product; and

a thread to cause the running thread counter to increment the number of running threads
and the standby thread counter to decrement the number of standby threads in response to a
run request, wherein the thread updates the product set by the product running thread counter if
the incremented number of running threads is greater than the product.

22. (CURRENTLY AMENDED) A computer readable storage medium storing a program for controlling at least one processor to execute a plurality of threads that perform parallel processing, according to a method comprising:

monitoring-counting a number of running threads performing parallel processing using a

running thread counter and a number of standby threads that are in a standby state using a standby thread counter for a predetermined time period;

setting a maximum number of running threads in accordance with the number of running threads during the predetermined time period using a maximum running thread counter;

comparing the number of standby threads with the maximum number of running threads;
and

terminating an amount of the standby threads exceeding the maximum number when the number of standby threads is greater than the maximum number;

incrementing the number of running threads using the running thread counter and decrementing the number of standby threads using the standby thread counter in response to a run request; and

updating the maximum number of running threads set by the maximum running thread counter if the incremented number of running threads is greater than the maximum number of running threads.

23. (CURRENTLY AMENDED) A computer readable storage medium storing a program for controlling at least one processor to execute a plurality of threads that perform parallel processing, according to a method comprising:

monitoring-counting a number of running threads performing parallel processing using a running thread counter and a number of standby threads that are in a standby state using a standby thread counter for a predetermined time period;

setting an average number of running threads in accordance with the number of running threads during the predetermined time period using an average running thread counter;

comparing the number of standby threads with the average number of running threads;
and

terminating an amount of the standby threads exceeding the average number when the number of standby threads is greater than the average number;

incrementing the number of running threads using the running thread counter and decrementing the number of standby threads using the standby thread counter in response to a run request; and

updating the average number of running threads set by the average running thread counter if the incremented number of running threads is greater than the average number of running threads.

24. (CURRENTLY AMENDED) A computer readable storage medium storing a program for controlling at least one processor to execute a plurality of threads that perform parallel processing, according to a method comprising:

~~monitoring~~ counting a number of running threads performing parallel processing using a running thread counter and a number of standby threads that are in a standby state using a standby thread counter for a predetermined time period;

setting a product obtained by multiplying the number of running threads during the predetermined time period by a predetermined coefficient using a product running thread counter;

comparing the number of standby threads with the product; ~~and~~

terminating an amount of the standby threads exceeding the product when the number of standby threads is greater than the product;

incrementing the number of running threads using the running thread counter and decrementing the number of standby threads using the standby thread counter in response to a run request; and

updating the product set by the product running thread counter if the incremented number of running threads is greater than the product.